II. REMARKS

A. Introduction

In the February 25, 2004 Office Action pending claims 1-10, 14 and 15 are rejected, and claims 11-13 and 16 are withdrawn.

In this Response, claims 1-10, 14 and 15 are canceled, news claims 17-20 are added, and Remarks are provided.

Support for the new claims can be found, e.g., at page 8, line 4 to page 12, line 15 of the application as filed.

B. New Claims

According to new claims 17-20, when a source code with a label is translated into a code including a memory address corresponding to the label, an address value for performing exception handling is set to the memory address in the code. When the (translated) code is executed, if the address value for performing exception handling is detected, the exception handling starts and performs reading out a real memory address value of a memory, which is used for executing the code, from a label table, and re-writing the address value in the code into the real memory address value. When the real memory address value in the code is detected during executing the code, the code is executed without performing the exception handling.

Thus, the memory address in the code is referred to during executing the code, and if the real memory address value is detected, executing the code is continued. On the contrary, if the address value for performing exception handling in the code is detected during executing the code, the exception handling to re-write the address value into the real memory address value is performed. Once the address value in the code is re-written into the real memory address value, in subsequent execution of the code with the real memory address, the exception handling step is omitted. Therefore, an execution speed of the program increases.

In the prior art, when a program with an address is executed, the address is checked whether the address is valid or invalid. If the address is invalid, address resolution processing is performed. The address-check-processing is performed whenever the program is executed. On the other hand, with the present invention, exception handling is performed instead of the

address-checking-processing. Therefore, when the exception handling does not start by re-writing the memory address value in the code, it is the same as when the address-check-processing is omitted. Therefore, an execution speed of the program increases relative to the prior art.

C. New Claims 17-20 vis-à-vis the Cited Art

Canceled claims 1-10 and 14-15 had most recently been rejected as being made obvious by a combination of Nakamura, Japanese Reference No. 5100864, and Gan et al., U.S. Patent No. 5,878,238. Most particularly, the Examiner indicated that Nakamura "teaches the converting or decoding of a program and rewriting an address described in the decoded program into an effective address" (Office Action, page 5, last two lines). However, the Examiner acknowledges that Nakamura at least does not expressly teach "exception handling", for which Gan et al.'s exception handler (see, e.g., Fig. 2 thereof) is cited.

For the following reasons, it is respectfully submitted that the present invention, as recited by new claims 17-20, was not rendered obvious by the cited combination.

Nakamura, an English translation of which is included in the Information Disclosure Statement filed concurrently herewith, merely discloses that a value (content) of a "second label part" included in a format of a source code is re-written into a value (content) off a "second label part" having highest priority (see paragraphs 0013 and 0015 in Nakamura). Nakamura does not teach or suggest the limitations included in each of new claims 18-20 of "translating a source code with a label into a code including a memory address corresponding to the label", and setting "the memory address in the code to an address value for performing exception handling".

Further, Nakamura re-writes a value of the "second label part" of the format of the source code (before translating). On the contrary, the present invention re-writes a memory address value in the code, which was translated from the source code by the exception handling.

Also a target of re-writing in Nakamura is a value (content) of a "label". On the contrary, the target of re-writing in the present invention is a value of a "memory address."

Thus, Nakamura does not teach or suggest at least the translating unit and the exception handling unit of the new claims, at least the latter of which feature is acknowledged in the Action.

Gan et al. merely discloses that a preset address table is referred to by exception handling. Gan et al. does not teach or suggest the claim 17 limitation:

> an exception handling unit starting only when said executing unit detects the address value for performing exception handling in the code during executing the code, and performing exception handling for re-writing the address value for making exception handling in the code into the real memory address value after reading out the real memory address value from said label table, based on the address value for performing exception handling or the label included in the source code, so that when said executing unit detects the real memory address value in the code, said executing unit executes the code, without performing the exception handling.

Claims 18-20 have similar recitations.

Further, Nakamura and Gant et al. do not teach or suggest that re-writing the memory address value in the code is performed by the exception handling to omit the address-check-processing performed every time in the prior art.

Thus, a combination of Nakamura and Gant et al. merely discloses that processing for re-writing a value of a "second label part" included in the source code is performed by exception handling.

III. CONCLUSION

In light of the above amendments and remarks, it is respectfully submitted that claims 17-20 are now in condition for allowance.

If there are any additional fees associated with this Response, please charge same to our Deposit Account No. 19-3935.

Finally, if there are any formal matters remaining after this Response, the undersigned would appreciate a telephone conference with the Examiner to attend to these matters.

Respectfully submitted,

STAAS & HALSEY LLP

Date: 6/25/04

By: _____
William F. Herbert
Registration No. 31,024

1201 New York Avenue, NW
Suite 700
Washington, DC 20005
202-434-1500